IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Eric L. Barsness, et al.	:	Date: October 13, 2010
Group Art Unit:	2161	:	IBM Corporation
Examiner:	Cindy Nguyen	:	Intellectual Property Law
Serial No.:	10/824,054	:	Dept. 917, Bldg. 006-1
Filed:	April 14, 2004	:	3605 Highway 52 North
Title:	DYNAMICALLY MANAGING COMPUTER RESOURCES BASED ON VALUATIONS OF WORK ITEMS BEING PROCESSED	:	Rochester, MN 55901

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 223313-1450

APPEAL BRIEF IN SUPPORT OF APPEAL
FROM THE PRIMARY EXAMINER TO THE BOARD OF APPEALS

Sir:

This is an appeal of a Final Rejection under 35 U.S.C. §102(e) of claims 6-9, 23-26, and 28-43 of Application Serial No. 10/824,054, filed April 14, 2004. This brief is submitted pursuant to a Notice of Appeal filed August 13, 2010, as required by 37 C.F.R. §41.31.

1. Real Party in Interest

International Business Machines Corporation of Armonk, NY, is the real party in interest. The inventors assigned their interest as recorded on April 14, 2004, on Reel 015227, Frame 0928.

2. Related Appeals and Interferences

There are no related appeals nor interferences pending with this application.

3. Status of Claims

Claims 6-9, 23-26, and 28-43 are pending and stand finally rejected, and are on appeal herein. Claims 1-5, 10-22, and 27 are cancelled. The claims on appeal are set forth in the Appendix of Claims.

4. Status of Amendments

Applicants submitted an Amendment following final rejection on July 13, 2010. This Amendment did not amend any claim, but included a statement regarding common ownership to obviate a potential rejection under 35 U.S.C. §103(a) and a response to the outstanding rejections. By Advisory Action dated July 27, 2010, the Examiner indicated that the Amendment did not place the application in condition for allowance and that the final rejections stand.

5. Summary of Claimed Subject Matter

The invention herein relates to allocation of resources for executing database queries according to work item valuation. Independent claims 6, 23, 29, 33, 37 and 41 recite respectively a method for managing access to computer resources, a method of providing fee-based processing, a data processing apparatus, a program product, a

networked environment, and a method for managing access to computer resources, which claim the following subject matter.[1]

In accordance with claim 6, a computer-implemented method for managing access to computer resources includes defining a respective financial valuation of each of multiple work items to be processed by one or more data processing systems [Specification ¶0029, 0035, 0039, 0040; Fig. 5, blocks 510, 512, 514]. The respective financial valuation of each work item is compared to a respective cost of accessing the additional computer resources needed to process it in a current time period, the additional resources being external to the data processing system(s) [Specification ¶0011, 0030, 0042; Fig. 6, block 614]. With respect to each work item for which the financial valuation exceeds the additional resource cost, the additional resources are dynamically accessed [Specification ¶0006, 0028, 0030, 0042, 0043; Fig. 4, block 418; Fig. 6, block 616]. With respect to each work item for which the financial valuation does not exceed the additional resource cost, the item is deferred to a subsequent time period [Specification ¶0030, 0042; Fig. 6, block 618]. The steps are repeated in one or more subsequent time periods with respect to the deferred items until each such work item has been processed [Specification ¶0030, 0043; Fig. 4, block 420].

In accordance with claim 23, a method is claimed of providing fee-based processing for programs in a processor system having at least one processor, a memory coupled to the processor, and a scheduling manager residing in the memory, the fees

[1] In the summary of claimed subject matter herein, appellant has cited selective portions of the specification and drawing showing each recited limitation in compliance with 37 C.F.R. 41.67(v). However, the fact that selective portions of the specification are referenced should not be construed to mean that support for the claimed limitations can not be found in other portions of the specification, or that other portions of the specification are not relevant or are less relevant to the disclosure of the invention herein.

based on utilization of computer resources for processing a program [Specification 0024, 0026, 0028; Fig. 1, feature 100; Fig. 2, features 130, 140, 143]. A respective financial valuation of each of multiple programs to be processed by one or more data processing systems is defined [Specification ¶0029, 0035, 0039, 0040; Fig. 5, blocks 510, 512, 514]. The respective financial valuation of each program is compared to a respective projected fee for utilizing the computer resources needed to process it in a current time period [Specification ¶0011, 0030, 0042; Fig. 6, block 614]. With respect to each program for which the financial valuation exceeds the projected fee, the computer resources are dynamically accessed [Specification ¶0006, 0028, 0030, 0042, 0043; Fig. 4, block 418; Fig. 6, block 616]. With respect to each program for which the financial valuation does not exceed the projected fee, the item is deferred to a subsequent time period [Specification ¶0030, 0042; Fig. 6, block 618]. The steps are repeated in one or more subsequent time periods with respect to the deferred programs until each such program has been processed [Specification ¶0030, 0043; Fig. 4, block 420].

In accordance with claim 29, a data processing system comprises at least one processor, a memory coupled to the at least one processor, and a scheduling manager residing in the memory and executable on the at least one processor [Specification 0026, 0028; Fig. 2, features 116, 130, 140, 143]. The scheduling manager dynamically manages access of each of multiple work items to additional computer resources external to the data processing system for processing the work item, each work item being performable by a data processing system and have a respective financial valuation [Specification ¶0028, 0029, 0035]. The scheduling manager compares, in each of multiple time periods, the respective financial valuation of each unprocessed work item to a respective cost of accessing the additional computer resources to process it in the respective time period [Specification ¶0011, 0030, 0042, 0043; Fig. 6, block 614]. With respect to each work

item for which the financial valuation exceeds the additional resource cost, the scheduling manager dynamically accesses the additional resources to process the work item in the respective time period [Specification ¶0006, 0028, 0030, 0042, 0043; Fig. 4, block 418; Fig. 6, block 616]. With respect to each work item for which the financial valuation does not exceed the additional resource cost, the scheduling manager defers processing of the work item to a subsequent time period [Specification ¶0030, 0042; Fig. 6, block 618].

In accordance with claim 33, a program product comprises a scheduling manager embodied as multiple computer-executable instructions recorded on a computer-readable storage medium and executable by a computer system [Specification 0028, 0045, 0046; Fig. 2, feature 143]. When executed, the scheduling manager causes the computer system to compare a respective defined financial valuation of each of multiple work items to be processed by the computer system to a respective cost of accessing the additional computer resources needed to process it in the a current time period, the additional resources being external to the computer system [Specification ¶0011, 0030, 0035, 0042, 0043; Fig. 6, block 614]. With respect to each work item for which the financial valuation exceeds the additional resource cost, the scheduling manager causes the computer system to dynamically access the additional resources to process the work item in the current time period [Specification ¶0006, 0028, 0030, 0042, 0043; Fig. 4, block 418; Fig. 6, block 616]. With respect to each work item for which the financial valuation does not exceed the additional resource cost, the scheduling manager causes the computer system to defer processing of the work item to a subsequent time period [Specification ¶0030, 0042; Fig. 6, block 618]. The scheduling manager causes the computer to repeated the steps in one or more subsequent time periods with respect to the deferred work items until each such work time has been processed [Specification ¶0030, 0043; Fig. 4, block 420].

In accordance with claim 37, a networked environment comprises a grid of computer resources and a request manager of the grid to receive requests of one or more customers for utilization of grid computing resources [Specification ¶0021, 0022, 0024; Fig. 1, features 100, 104, 108]. The networked environment further comprises one or more computer systems of a customer coupled to the request manager, the computer systems having one or more processors, a memory coupled to the at least one processor, and a scheduling manager residing in the memory and executable on the at least one processor [Specification 0026, 0028; Fig. 1, Fig. 2, features 116, 130, 140, 143]. The scheduling manager dynamically manages access of each of multiple work items to additional computer resources external to the data processing system for processing the work item, each work item being performable by a data processing system and have a respective financial valuation [Specification ¶0028, 0029, 0035]. The scheduling manager compares, in each of multiple time periods, the respective financial valuation of each unprocessed work item to a respective cost of accessing the additional computer resources to process it in the respective time period [Specification ¶0011, 0030, 0042, 0043; Fig. 6, block 614]. With respect to each work item for which the financial valuation exceeds the additional resource cost, the scheduling manager dynamically accesses the additional resources to process the work item in the respective time period [Specification ¶0006, 0028, 0030, 0042, 0043; Fig. 4, block 418; Fig. 6, block 616]. With respect to each work item for which the financial valuation does not exceed the additional resource cost, the scheduling manager defers processing of the work item to a subsequent time period [Specification ¶0030, 0042; Fig. 6, block 618].

In accordance with claim 41, a computer-implemented method for managing access to computer resources includes providing multiple work items for processing by one or more data processing systems in a current time period, each work item having a

respective financial valuation [Specification ¶0029, 0035, 0039, 0040; Fig. 4, blocks 404-407; Fig. 5, blocks 510, 512, 514]. A first subset of the multiple work items is selected for processing by a first data processing system in the current time period according to the financial valuation [Specification ¶0036, 0041, 0042; Fig. 4, blocks 408, 414, 415, 416, Fig. 6, blocks, 608, 610, 612]. With respect to each work item not included in the first subset, the respective financial valuation of the work item is compared to a respective cost of accessing additional computer resources external to the first data processing system to process it in the current time period [Specification ¶0011, 0030, 0042; Fig. 6, block 614]. With respect to each work item not included in the first subset for which the financial valuation exceeds the respective cost of accessing additional resources, the additional computer resources are dynamically accessed to process the work item in the current time period [Specification ¶0006, 0028, 0030, 0042, 0043; Fig. 4, block 418; Fig. 6, block 616]. With respect to each work item not included in the first subset for which the financial valuation does not exceed the respective additional resource cost, the item is deferred to a subsequent time period [Specification ¶0030, 0042; Fig. 6, block 618]. The steps are repeated in one or more subsequent time periods, wherein any work item deferred is included in the multiple work items of each subsequent time period until the work item is processed [Specification ¶0030, 0043; Fig. 4, block 420]. For at least some time periods, the first subset includes fewer than all of the respective multiple work items. [Specification ¶0036, 0042; Fig. 4, block 408, Fig. 6, block 608].

6. Grounds of Rejection To Be Reviewed on Appeal

Claims 6-9, 23-26 and 28-43 are finally rejected under 35 U.S.C. §102(e) as anticipated by Clarke et al.(U.S. Patent Publication 2004/0221038). The only issues in this appeal are whether the claims are anticipated by *Clarke*.

7. Argument

Appellants contend that the Examiner failed to establish adequate grounds of rejection for the following reasons:

I. The Examiner improperly rejected claims 6-9, 23-26 and 28-43 under 35 U.S.C. §102(e) because *Clarke* does not disclose key claim limitations, specifically associating a respective financial valuation with each of multiple work items to be processed, comparing this valuation to cost of resources, and selectively accessing resources based on that comparison. [page 11 below].

II. Obviousness is not a basis for rejection of the claims, because *Clarke* was commonly owned [page 19 below].

Overview of Invention

A brief overview of appellants' invention in light of existing art will be helpful in appreciating the issues herein. Appellants' invention relates to the dynamic management of computer resources, and in particular to scheduling items for processing and dynamically obtaining resource for processing based on financial justification.

Certain computing environments exist wherein the available amount of computer processing resources are flexible, and depend to some degree on one's willingness to pay for them. For example, it is known to provide computer processing resource via a

computing grid which is accessed via a network, which can be purchased on demand. The computing grid is a collection of processors or computer systems, to which access is managed by some form of manager. The grid may be considered a server which performs computing services on behalf of clients. These services are generally provided for a fee, which may be based on number of processor cycles or some appropriate measure. I.e., there is a direct, financial cost associated with their use. The fees for computing resources could vary by time of day and/or day of week, or according to how busy the system is or other factors

In an exemplary embodiment, a client connected to the grid is a computer system or collection of systems within an enterprise, having some internal capability to process items of work. The workload of items which the client is required to process may vary considerably, and may sometimes exceed the capability of the client to internally process it. In such a case, the client may access the grid or other external resource to provide the necessary processing capability to process the work items. However, because there is generally some cost associated with use of a grid or other additional resource, it may be best in some cases to simply defer the work item to a later time period, and process it in the client system without incurring the additional cost.

Although various conventional techniques exist for prioritizing tasks and allocating resources in a data processing environment, appellants' invention is directed to a very specific application, in which a decision must be made whether to acquire fee-based resources to perform specific tasks in a current time period, as opposed to deferring the tasks to a later time period, based on economic justification. Conventionally, insufficient consideration has been given to how and under what circumstances to acquire additional computing resources needed to do the work now. Such a decision may be

made manually, or may be on the basis of simple fixed rules. For example, additional resources may be accessed whenever internal resources are insufficient.

Appellants recognized that it would be desirable to provide an automated system for determining whether or not to access additional processing resources and incur the associated cost, and in particular one which takes into account the financial valuation of processing the work item immediately and compares it to the cost of additional resources to do so. Accordingly, appellants disclose a technique in which a respective financial valuation is associated with each of multiple work items to be processed. This financial valuation is intended to represent a value, in monetary terms, associated with having the task done now as opposed to later. The items with highest valuation are processed with available internal resources. With respect to the remaining items, the financial valuation of each such item is compared to the cost of additional resources needed to process it in the current time interval. If the valuation exceeds the cost, the additional resource is obtained and the work item is processed; otherwise the item is deferred to a later time. The comparison of financial valuation to cost is on an item-by-item basis, enabling some items (the more "valuable" ones) to be processed by acquiring additional resources at a cost, while others are deferred. Appellants' system thus permits a direct comparison of financial valuation of performing work items to cost of processing in a current time interval, providing a more accurate and more intelligent automated determination with respect to acquiring additional processing resources.

Appellants' claimed invention is not a high-level concept of taking financial considerations into account when allocating resources. It is, rather, a specific application of that general principle to an environment in which additional resources are available on demand at an additional cost, and a variable number of work items having varying

financial valuations are waiting to be processed. Among the significant features recited in appellants' independent claims is that a ***respective financial valuation*** is associated with ***each of multiple work items (or programs) to be processed*** by a computer system in a time interval, that this ***respective financial valuation is compared to a respective cost*** of resources needed to process the work item (or program) in that time interval, and that the ***resource is selectively obtained based on that comparison***. These features are not disclosed in the cited reference.

I. The Examiner improperly rejected claims 6-9, 23-26 and 28-43 under 35 U.S.C. §102(e) because *Clarke* does not disclose key claim limitations, specifically associating a respective financial valuation with each of multiple work items to be processed, comparing this valuation to cost of resources, and selectively accessing resources based on that comparison

In order to support a rejection for anticipation, each and every element of the claimed invention must be shown in a single prior art reference. Appellants' claims are not anticipated by *Clarke* because *Clarke* does not disclose, inter alia, associating a respective financial valuation with processing each of multiple work items (or programs), or comparing this respective financial valuation to a respective cost of computer resources to process each work item in a current time interval, or selectively accessing computer resources for processing each respective work item based on the comparison. All appellants' independent claims recite these essential features.

Appellants' representative independent claim 6 recites:

6. A computer-implemented method for managing access to computer resources, the method comprising:

(a) *defining a respective financial valuation of each of a plurality of work items to be processed* by one or more data processing systems;

(b) *comparing the respective financial valuation of each respective said work item to a respective cost* of accessing additional computer resources necessary *to process the work item in a current time period*, said additional computer resources being external to said one or more data processing systems;

(c) with respect to each said work item for which the *respective financial valuation of the work item exceeds the respective cost* of accessing additional computer resources necessary to process the work item in the current time period, *dynamically accessing additional computer resources* necessary to process the work item in the current time period;

(d) with respect to each said work item for which the *respective financial valuation of the work item does not exceed the respective cost* of accessing additional computer resources necessary to process the work item in the current time period, *deferring processing of the work item* to a subsequent time period; and

(e) repeating said (b) through (d) in one or more subsequent time periods with respect to each said work item deferred by said (d) until each said work item has been processed. [emphasis added]

The remaining independent claims vary in scope, but all contain limitations analogous to the italicized limitations above[2].

Clarke discloses a system for configuring a distributed processing computing environment. In accordance with *Clarke*, a distributed computing system is configured so as to maximize its value determined in accordance with some value criteria. In particular, a preferred value criterion for the distributed system is "return on investment" (ROI). In other words, one or more valuation metrics are established for various hardware and

[2] Independent claim 23 does not recite "additional computer resources", but recites a method of providing fee-based processing, in which the financial valuation is compared with the projected fee for utilization of computer resources. Various other differences exist, but all claims recite associating financial valuations with work items or programs to be processed, making comparisons of these valuations to costs or fees, and accessing resources based on the comparisons.

software components of a system, a system value is determined in response to these metrics, and elements of a distributed computing environment are re-configured according to the determined valuations.

Clarke's basic algorithm is shown in *Clarke*'s Fig. 3. As shown in Fig. 3, at least one system metric is determined, a "value" of a "computing environment" is determined, and a determination is made based on the value whether to change the computing environment.

Clarke discloses that a large variety of factors may influence the overall determination of a system or environment value. These may include, e.g., internal corporate financial data such as cost of labor, real estate, electric power, penalties for non-compliance with service contracts, taxes, free cash flow, etc..; and externally provided data such as interest rates, contractor labor rated, cost of public compute capacity, prices of server hardware, etc.[3] In other words, the data on which *Clarke* bases valuation determinations could be almost anything under the sun that might affect value of the system configuration. Among the factors disclosed is a value associated with applications in progress in the system.

Although *Clarke* discloses the high-level concept of taking financial considerations into account in managing a computer system, they do not disclose the particular application of the general concept which is disclosed and claimed herein by appellants. Ultimately, *Clarke* is assigning a value to a "distributed computing environment" for the purpose of maximizing the value of the "distributed computing environment". It is true that applications in progress constitute one of the components of

[3] See, e.g., *Clarke*, paragraphs 0041, 0044.

that value, but there is no individual value associated with work items to be processed, or comparison of such values with cost of resources to process the work items.

Specifically, *Clarke* does not disclose "*...**defining a respective financial valuation of each of a plurality of work items to be processed**...*", as recited in claim 6. The Examiner cites the following passage from *Clarke* for the requisite teaching:

> [0044] The externally provided data, represented as data 250 may be obtained over a network, from a number of sources, including, for example, but not limited to: sources accessible over the world wide web, from a third party service provider, or from a corporate function. The data may include but is not limited to: financial data such as generally available interest rates, contractor labor rates, cost of public compute capacity, price of servers to be bought from a dealer, price of on-demand business services, marketplace information for buyers and sellers of IT resources, etc. Further the data may include, but is not limited to: IT information such as jobs available for servicing, offered prices, and required resource configurations (e.g., jobs requiring resources), available web services and prices, etc. The data is shown as being provided to the aggregation point 240, but may be provided directly to the value determining element 260, change determining element 270, or aggregated with other data in a sub-aggregating element.

While the above passage does indeed mention "jobs available for servicing", this is only in the context of a large list of data which can be fed into a value determining element. The value determining element does not determine "a respective financial value of ***each of a plurality of work items to be processed***", but determines an overall value of a "distributed computing environment". The above quoted passage does not even state that some intermediate value is determined for each of multiple jobs; all it states is that "jobs available for servicing" is one of many potential input variables in a complex value determining function. To read anything else into this passage amounts to extracting from appellants' disclosure and applying it to fill in what *Clarke* does not say.

At best. the above passage from *Clarke*, which the Examiner cites as a teaching of the applicable claim limitation, only discloses the high level idea of using of financial data and making value determinations. It does not disclose the specific claim limitation.

Although not cited by the Examiner, appellants would point out that several later passages in *Clarke* disclose assigning a respective value to each of multiple applications *in progress*, based in part on the percentage of completion of each application.[4] This again is input data to a complex value function for determining the value of the distributed processing environment. The computation is performed because an "environment change" may include cancelling work in progress, i.e., losing the value of such work.[5] However, these passages do not meet the applicable claim limitation, because the claim recites a value of each of multiple "work items ***to be processed***", not of work items already processed.

Nor does *Clarke* disclose ***"...comparing the respective financial valuation of each respective said work item to a respective cost of accessing additional computer resources necessary to process the work item in a current time period..."***, as recited in claim 6. The Examiner cites several passages from *Clarke* to support his read, but collectively they show only the general use of financial considerations for configuration determinations, and do no show the specific claim limitation. The primary passage cited by the Examiner discloses essentially that a "change determining process" uses values determined by a "value determining process" to determine whether a change is advisable to a "distributed computing environment", and is quoted in full below:

---

[4] See, e.g., *Clarke* ¶[0064]

[5] *Clarke*, ¶[0065]

[0046] Further depicted in FIG. 2 is the value determining process depicted as element 260. This element may be co-located with the IT resources of element 210 or, may be located elsewhere. It may be operated by the enterprise or by a consortium, or a third party service provider. The value determining process may use some or all the data provided, may include estimated values, may perform algorithmic calculations, and may request additional data from one of the preceding sources, or from additional external sources. It may be combined with element 270, the change determining process. That is, responsive to the value(s) determined by element 260, is element 270 which implements a change determining process for determining whether a change in the environment is advisable. Change determining may include, but is not limited to: performing optimization algorithms, performing multiple value calculations, negotiating penalties with third parties (e.g., attempting to reduce potential SLA penalties through electronic or non-electronic negotiation communication). Change determining may be real-time, near real-time or non real-time. It may include notification of personnel for review, or personal negotiation with an external or internal partner. Change determining may include processes for determining and specifying an effective time period during which changes should be in effect, and specifying a process or person to monitor subsequent to the change. Although not shown, it is understood that a process is provided for effecting any change that is determined and this process for effecting a change may be partly, or fully automated.

The Examiner further cites passages disclosing that responsive to metrics collected and a determined value, a determination is made whether a change in the distributed computing environment is advisable, alternate values being based on potential changes to workload and configuration;[6] that various data may be used to assign a value

[6] *Clarke*, ¶[0055], which provides:

Continuing to step 330 in FIG. 2, responsive to the metrics collected in step 310, and the value determined in step 320, a determination is made as to whether a change in the distributed computing environment is advisable. In a preferred embodiment, alternate values are determined based on potential changes tot he distributed computing workload (e.g., addition or deletion of particular application jobs at particular locations), configuration modifications (e.g., addition or removal or resources from a grid), and the resultant economic consequences of increased revenue, decreased cost, and potential SLA penalties. If the alternate values indicate a higher value may be obtained, then the decision is made to alter the environment. Note that the change determination process may be provided as a service by a 3rd party or

to an application in progress[7]; that multiple potential change are evaluated and a change is selected according to feasibility[8]; and that IT data may include, among other things, data representative of external IT resources.[9]

The problem is that all of these things are very general, and appellants' claims recite something specific. Appellants will concede that what the claims recite, i.e. *"...comparing the respective financial valuation of each respective said work item to a respective cost of accessing additional computer resources necessary to process the work item in a current time period...."*, might be considered a specific example of the general concept of using financial data to make a decision affecting system configuration. But it is an erroneous method of analysis to conclude that the specific is disclosed by the general. Anticipation requires a disclosure of the specific things claimed by appellant, not a disclosure of concepts sufficiently broad to encompass the specific claimed features (and a whole lot of other things as well).

The Examiner has cited numerous passages from *Clarke*, collectively amounting to several columns of text, which disclose in essence that a great variety of data may be combined to determine valuations of a distributed computing environment, and that these value determinations may be used to determine changes to be made to the configuration of the distributed computing environment. There is nothing in all that text which

---

may be operated by the owner of the distributed computing environment.

[7] *Clarke*, ¶[0063]

[8] *Clarke*, ¶[0068]

[9] *Clarke*, ¶[0043]

discloses the claimed limitation, i.e. that a specific item of value information (respective financial valuation of each of multiple work items to be processed) is compared to a specific item of cost information (cost of accessing additional computer resources needed to process the work item in a current time period).

Finally, *Clarke* fails to disclose "***...dynamically accessing additional computer resources necessary to process the work item in the current time period...***" responsive to the previous value vs. cost determination. As in the case of the previous recitations, *Clarke* discloses the general concept of altering a distributed computing environment configuration, and in particular this could include "addition or removal of resources from a grid".[10]. But this amounts to a re-configuration of existing resources (the grid), not an acquisition of additional resource, and is performed responsive to a system value determination, i.e, that the value of the distributed computing environment is enhanced by such an alteration. There is no nexus disclosed in *Clarke* between these resources added or removed from a grid and specific work items which were compared in the previous step.

Clarke's "distributed processing environment" is apparently a distributed system owned entirely by some single enterprise. Although this is indeed disclosed as a grid, Clarke does not disclose obtaining additional resources external to the enterprise at a cost or fee. Clarke is concerned entirely with maximizing the "value:", i.e. return on investment, to the owner of the distributed processing system, which could be. the server or provider of on-demand computing services. Appellants, on the other hand, disclose and claim the peripherally related technique of determining, ***in the client***, whether and for what work items to acquire additional on-demand computing resources.

---

[10] *Clarke* ¶[0055], [0056]

For all of the above reasons, *Clarke* fails to disclose one or more recited limitations of the independent claims, and the rejection for anticipation was erroneous.

II. Obviousness is not a basis for rejection of the claims, because *Clarke* was commonly owned.

Although the Examiner did not explicitly reject the claims as obvious over *Clarke*, an anticipation rejection may be deemed to include an implied or "subsumed" obviousness rejection, which the appellants will briefly address.

Appellants have previously submitted on the record a "Statement Regarding Common Ownership"[11] to establish common ownership by International Business Machines Corporation at the time the present invention was made. Accordingly, *Clarke* is not available as a reference to support an obviousness rejection under 35 U.S.C. §103.

Appellants submit that the Congressional amendment to 35 U.S.C. §103(c), which added references under 35 U.S.C. §102(e) to the scope of the exception expressed therein, is worth noting. Congress clearly expressed an intent to prevent companies from being penalized by citing their own pending patent applications as showing "obviousness", since it is well known that a large company may file multiple patents arising out of research and development in related areas.

The line drawn by Congress is clear. The Examiner may base a rejection on a commonly assigned application as an anticipatory reference which discloses each and

[11] This statement is contained in the paper filed July 13, 2010.

every claim limitation, but may not draw obvious inferences from such an application where the explicit disclosure of each and every claimed element is not present.

Appellants do not necessarily concede that their claims would have been obvious in view of *Clarke*, but make this point to counter an apparent readiness on the part of the Examiner to fill in what *Clarke* does not explicitly disclose. ***Such a tendency is plainly contrary to the will of Congress as expressed in the amendment to 35 U.S.C. §103(c).*** The Examiner is not entitled to draw "obvious" inferences from *Clarke*'s disclosure, and must find ***each and every claim limitation*** expressly disclosed in *Clarke*. Anything less would subvert the intentions of Congress in amending §103(c). For the reasons stated earlier, one or more claim limitations are not disclosed in *Clarke*, and the rejections were therefore erroneous.

8. Summary

Appellants disclose and claim a novel technique for scheduling work items and allocating computing resources in an environment where additional resources are available on demand at a cost, wherein a respective financial valuation is associated with each of multiple work items to be processed, this valuation is compared with a cost of additional resources for processing in a time interval, and the resources are dynamically obtained based on the results of the comparison or the work item is deferred for a subsequent time interval. *Clarke*, the sole reference, discloses a technique for managing the configuration of a distributed computing environment, in which a variety of data is used to assign a respective value to different configurations and configuration changes are selected based on the valuation. Although *Clarke* relates generally to the use of financial considerations in making resource allocation decisions and discloses that workload or

applications are elements of a larger valuation determination for the system as a whole, for the reasons explained herein it contains no explicit disclosure of the specific elements recited by appellants in their claims.

For all the reasons stated herein, the rejections for anticipation were improper, and appellants respectfully request that the Examiner's rejections of the claims be reversed.

Date: October 13, 2010

Respectfully submitted,

ERIC L. BARSNESS, et al.

By_____

Roy W. Truelson, Attorney
Registration No. 34,265
(507) 202-8725 (Cell) (507) 289-6256 (Office)

From: IBM Corporation
Intellectual Property Law
Dept. 917, Bldg. 006-1
3605 Highway 52 North
Rochester, MN 55901

APPENDIX OF CLAIMS

1 - 5. (Cancelled)

6. A computer-implemented method for managing access to computer resources, the method comprising:

(a) defining a respective financial valuation of each of a plurality of work items to be processed by one or more data processing systems;

(b) comparing the respective financial valuation of each respective said work item to a respective cost of accessing additional computer resources necessary to process the work item in a current time period, said additional computer resources being external to said one or more data processing systems;

(c) with respect to each said work item for which the respective financial valuation of the work item exceeds the respective cost of accessing additional computer resources necessary to process the work item in the current time period, dynamically accessing additional computer resources necessary to process the work item in the current time period;

(d) with respect to each said work item for which the respective financial valuation of the work item does not exceed the respective cost of accessing additional computer resources necessary to process the work item in the current time period, deferring processing of the work item to a subsequent time period; and

(e) repeating said (b) through (d) in one or more subsequent time periods with respect to each said work item deferred by said (d) until each said work item has been processed.

7. The method of claim 6 further comprising applying a valuation heuristic to each work item.

8. The method of claim 6 further comprising applying a priority algorithm for preventing starvation of computer resources to those work items which have been delayed, whereby the processing of all the work items in a program is completed.

9. The method of claim 7 further comprising having the priority algorithm increase respective valuations of delayed work items so as to complete processing of each of the work items prior to or at a cut-off processing date of the work item.

10 - 22. (Cancelled)

23. A method of providing fee-based processing for programs in a processor system, whereby fees are based on utilization of computer resources for completing processing a program, the processor system including at least one processor; a memory coupled to the at least one processor, and a scheduling manager residing in the memory, the method comprising the steps of:

(a) defining a respective financial valuation of each of a plurality of programs to be processed;

(b) comparing the respective financial valuation of each respective said program to a respective projected fee for utilization of computer resources to process said program in a current time period;

(c) with respect to each said program for which the respective financial valuation of the program exceeds the respective projected fee for utilization of computer resources

to process the program in the current time period, dynamically accessing computer resources to be applied to process the program in the current time period;

(d) with respect to each said program for which the respective financial valuation of the program does not exceed the respective projected fee for utilization of computer resources to process the program in the current time period, deferring processing of the program to a subsequent time period; and

(e) repeating said (b) through (d) in one or more subsequent time periods with respect to each said program deferred by said (d) until each said program has been processed; and

(f) assessing a fee for the dynamically accessed computer resources to be used.

24. The method of claim 23 further comprising applying a valuation heuristic to each work item for establishing the valuation of each work item.

25. The method of claim 24 further comprising applying a priority algorithm for preventing starvation of computer resources to those work items which have been delayed, whereby the processing of all the work items in a program is completed.

26. The method of claim 25 wherein the dynamic determination is based on different attributes of the one or more work items forming at least part of a program.

27. (Cancelled)

28. The method of claim 6, wherein said method is used in a networked environment including a grid of computing resources, and a request manager of the grid to receive requests of one or more customers for utilization of computing resources of the grid;

wherein said additional computer resources comprise computing resources of said grid of computing resources; wherein one or more computer systems of a customer is coupled to the request manager and include one or more processors; a memory coupled to at least the one processor; and, a scheduling manager residing in the memory and executable by the at least the one processor.

29. A data processing apparatus comprising:

at least one processor;

a memory coupled to the at least one processor; and

a scheduling manager residing in the memory and executable on the at least one processor, the scheduling manager dynamically managing access of each of a plurality of work items to additional computer resources external to said data processing apparatus for processing the respective work item, each said work item being a respective item of work performable by a data processing system and having a respective financial valuation;

wherein said scheduling manager, in each of a plurality of time periods, compares the respective financial valuation of each unprocessed work item to a respective cost of accessing said additional computer resources to process the work item in the respective time period, and with respect to each said work item for which the respective financial valuation exceeds the respective cost of accessing the additional computer resources to process the work item in the respective time period, dynamically accesses the additional computer resources to process the work item in the respective time period; and with respect to each said work item for which the respective financial valuation does not exceed the respective cost of accessing the additional computer resources to process the work item in the respective time period, defers processing of the work time to a subsequent time period.

30. The apparatus of claim 29 wherein the scheduling manager applies a valuation heuristic to each work item.

31. The apparatus of claim 29 wherein the scheduling manager applies a priority algorithm for preventing starvation of computer resources to those work items which have been deferred, whereby the processing of all the work items is completed.

32. The apparatus of claim 31 wherein the priority algorithm increases respective valuations of delayed work items so as to complete processing of each of the work items prior to or at a cut-off processing date of the work item.

33. A program product comprising:

a scheduling manager embodied as a plurality of computer-executable instructions recorded on a computer-readable storage medium, wherein said scheduling manager, when executed by a computer system, causes the computer system to:

(a) compare a respective defined financial valuation of each of a plurality of work items to be processed by the computer system to a respective cost of accessing additional computer resources necessary to process the work item in a current time period, said additional computer resources being external to said computer system;

(b) with respect to each said work item for which the respective financial valuation of the work item exceeds the respective cost of accessing additional computer resources necessary to process the work item in the current time period, dynamically accesses additional computer resources necessary to process the work item in the current time period;

(c) with respect to each said work item for which the respective financial valuation of the work item does not exceed the respective cost of accessing additional computer resources necessary to process the work item in the current time period, defers processing of the work item to a subsequent time period; and

(d) repeats said (a) through (c) in one or more subsequent time periods with respect to each said work item deferred by said (c) until each said work item has been processed.

34. The program product of claim 33 wherein the scheduling manager applies a valuation heuristic to each work item to establish a valuation for each of the work items.

35. The program product of claim 33 wherein the scheduling manager applies a priority algorithm for preventing starvation of computer resources to those work items which have been delayed, whereby the processing of all the work items in a program will be completed.

36. The program product of claim 35 wherein the priority algorithm increases respective valuations of delayed work items so as to complete processing of each of the work items prior to or at a cut-off processing date of the work item.

37. A networked environment, comprising:

a grid of computing resources;

a request manager of the grid to receive requests of one or more customers for utilization of computing resources of the grid;

one or more computer systems of a customer coupled to the request manager; the one computer system comprising one or more processors;

a memory coupled to at least the one processor of the one computer system; and,

a scheduling manager residing in the memory and executable on the at least one processor, the scheduling manager dynamically managing access of each of a plurality of work items to additional computer resources external to said one or more computer systems of a customer for processing the respective work item, each said work item being an item of work performable by a data processing system and having a respective financial valuation;

wherein said scheduling manager, in each of a plurality of time periods, compares the respective financial valuation of each unprocessed work item to a respective cost of accessing said additional computer resources to process the work item in the respective time period, and with respect to each said work item for which the respective financial valuation exceeds the respective cost of accessing the additional computer resources to process the work item in the respective time period, dynamically accesses the additional computer resources to process the work item in the respective time period; and with respect to each said work item for which the respective financial valuation does not exceed the respective cost of accessing the additional computer resources to process the work item in the respective time period, defers processing of the work time to a subsequent time period.

38. The environment of claim 37 wherein the scheduling manager applies a valuation heuristic to each work item.

39. The environment of claim 37 wherein the scheduling manager applies a priority algorithm for preventing starvation of computer resources to those work items which have been delayed, whereby the processing of all the work items in a program is completed.

40. The environment of claim 39 wherein the scheduling manager increases respective valuations of delayed work items so as to complete processing of each of the work items prior to or at a cut-off processing date of the work item.

41. A computer-implemented method for managing access to computer resources, the method comprising:

(a) providing a plurality of work items for processing by one or more data processing systems in a current time period, each work item having a respective financial valuation;

(b) selecting a first subset of said plurality of work items for processing by a first data processing system in the current time period according to said financial valuations;

(c) with respect to each said work item not included in said first subset, comparing the respective financial valuation of the work item to a respective cost of accessing additional computer resources external to said first data processing system to process the work item in the current time period;

(d) with respect to each said work item not included in said first subset for which the respective financial valuation of the work item exceeds the respective cost of accessing additional computer resources external to said first data processing system to

process the work item in the current time period, dynamically accessing additional computer resources external to said first data processing system to process the work item in the current time period;

(e) with respect to each said work item not included in said first subset for which the respective financial valuation of the work item does not exceed the respective cost of accessing additional computer resources external to said first data processing system to process the work item in the current time period, deferring processing of the work item to a subsequent time period; and

(f) repeating said (a) through (e) in multiple time periods, wherein any work item deferred by (e) is included in the plurality of work items of each subsequent time period until the work item is processed, and wherein for at least some time periods, the first subset of the respective plurality of work items includes fewer than all of the respective plurality of work items.

42. The method of claim 41 further comprising applying a priority algorithm for preventing starvation of computer resources to those work items which have been deferred, whereby the processing of all the work items is completed.

43. The method of claim 41, wherein said method is used in a networked environment including a grid of computing resources, and a request manager of the grid to receive requests of one or more customers for utilization of computing resources of the grid; wherein said additional computer resources comprise computing resources of said grid of computing resources.

APPENDIX OF EVIDENCE

No evidence is submitted.

APPENDIX OF RELATED PROCEEDINGS

There are no related proceedings.